BAB IV
ANALISA DAN PERANCANGAN

4.1 Analisa

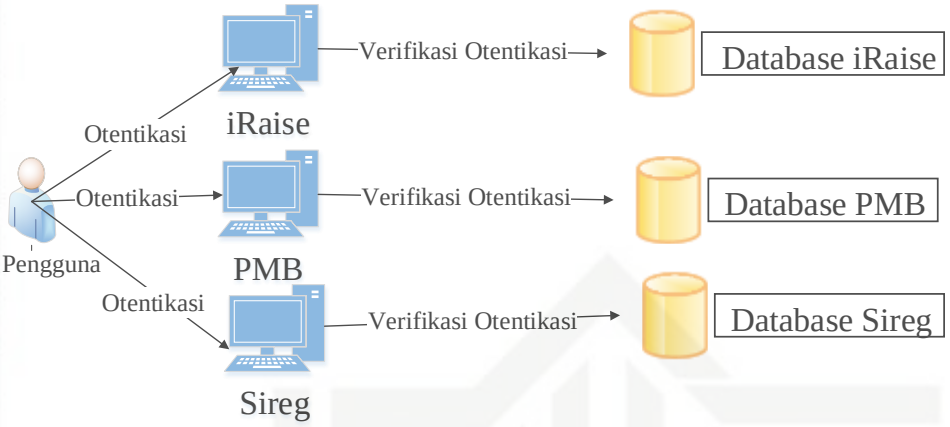
Tahap analisa dilakukan untuk menganalisa kebutuhan sistem yang akan dibuat. Pada tahap analisa, akan dilihat kebutuhan sistem dan data yang digunakan. Selain itu pada tahap ini digunakan untuk membuat alur proses dari sistem lama dan pengembangan sistem berbasis SSO.

4.1.1 Analisa Sistem Lama

Analisa sistem lama dilakukan untuk melihat alur sistem yang terjadi pada sistem yang sudah digunakan sebelumnya. Pada analisa sistem lama akan didapatkan permasalahan yang ada dan selanjutnya akan diselesaikan pada sistem baru. Berikut merupakan analisa sistem lama.

Pada kampus terdapat beberapa portal yang digunakan untuk menunjang proses bisnis yang ada. Pada Universitas Islam Negeri Sultan Syarif Kasim Riau sendiri memiliki beberapa portal yang digunakan oleh mahasiswa, dosen dan pegawai kampus. Pada masing-masing portal tersebut, pengguna memiliki akun masing-masing dengan *username* dan *password* yang berbeda pada setiap portalnya. Selanjutnya *username* dan *password* tersebut disimpan pada *database* yang berbeda sehingga setiap pengguna akan memiliki banyak *username* dan *password* yang digunakan untuk mengakses portal-portal tersebut.

Berdasarkan sistem lama tersebut, permasalahan yang timbul adalah banyaknya *username* dan *password* dari beberapa portal ini untuk setiap pengguna, sehingga setiap pengguna harus menginputkan *username* dan *password* yang berbeda untuk masing-masing portal. Selain itu masalah yang timbul adalah kemungkinan terjadi *human error* jika pengguna lupa terhadap *username* dan *password* yang berbeda, sehingga setiap pengguna harus mengingat *username* dan *password* yang berbeda agar bisa masuk ke masing-masing portal. Adapun alur proses otentikasi sistem lama dapat dilihat pada Gambar 4.1 berikut:



Gambar 4.1 Alur Proses Otentikasi Sistem Lama

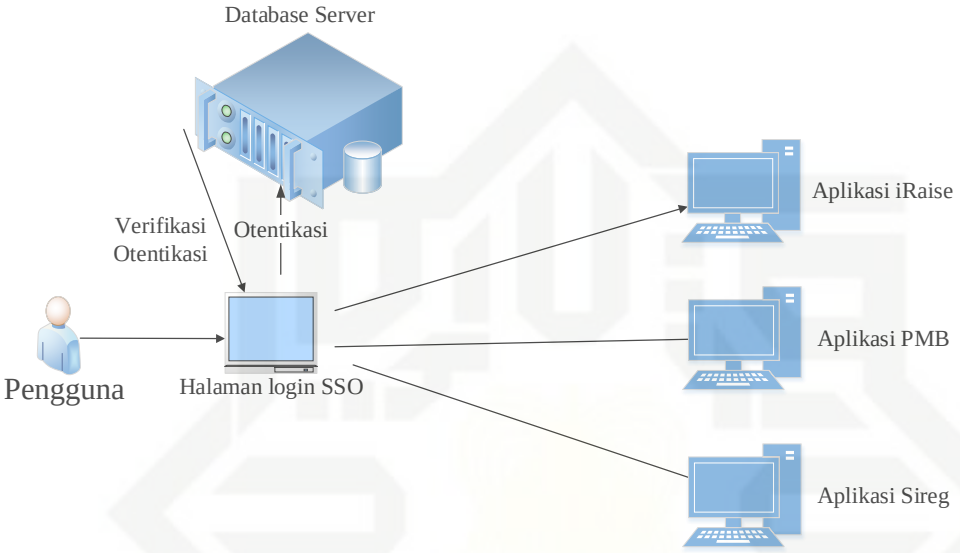
4.1.2 Analisa Pengembangan Sistem Berbasis SSO

Pada tahap analisa sistem baru dilakukan untuk melihat alur sistem baru yang akan dibuat. Pada sistem baru yang akan dibuat permasalahan pada sistem lama diharapkan akan dapat diselesaikan dengan menggunakan metode yang ada. Berikut merupakan analisa dari sistem baru.

Pada sistem baru yang akan dibuat, pengguna hanya 1 kali dalam menginputkan *username* dan *password* untuk semua akun portal kampus yang sudah terintegrasi. Sehingga pengguna tidak perlu memiliki banyak *username* dan *password* untuk portal yang berbeda. Hal ini bertujuan untuk mengurangi *human error* yang terjadi akibat pengguna yang lupa terhadap *username* dan *password* untuk masing – masing akunnya. Selain itu dengan hanya memiliki satu *username* dan *password* untuk semua portal akan mempermudah dalam pengelolaan data *user*.

Sistem baru yang akan dibuat menggunakan *Single Sign On* (SSO) yang berfungsi untuk membantu pengguna atau *user* masuk ke dalam beberapa aplikasi hanya dengan sekali proses otentikasi atau proses *input username* dan *password*. Penggunaan SSO ini akan sangat membantu dalam proses bisnis yang besar dalam banyak aplikasi dikarenakan *user* harus menginputkan *username* dan *password* dan hal tersebut dapat dilakukan dengan satu kali proses *input* sehingga *user* dapat mengakses banyak aplikasi. Selain memudahkan pengguna dengan hanya sekali

memasukkan *username* dan *password*, penggunaan SSO dalam proses bisnis yang besar terutama dalam lingkungan kampus dapat memudahkan penyimpanan data pengguna yang hanya membutuhkan 1 tabel *user* dalam *database server* yang befungsi untuk menyimpan data *username* dan *password* pengguna.



Gambar 4.2 Analisa Pengembangan Sistem Berbasis SSO

4.2 Perancangan

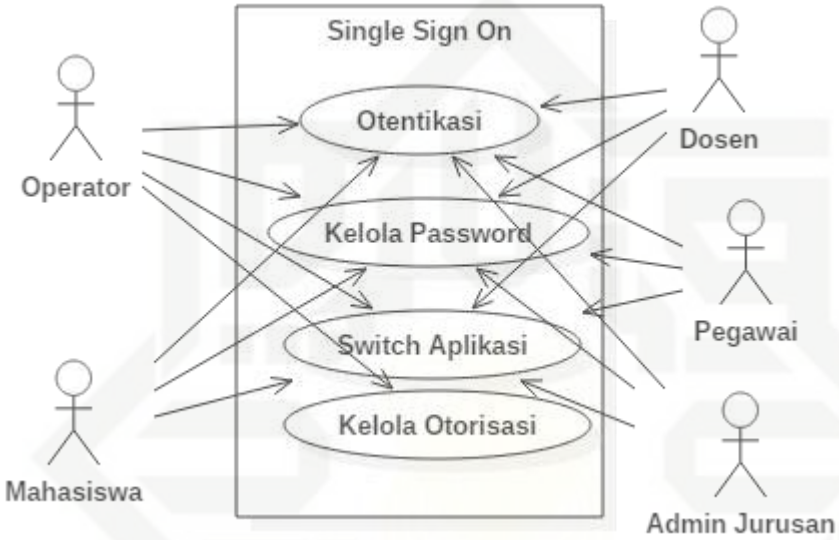
Perancangan bertujuan untuk menampilkan alur dan kebutuhan sistem dalam beberapa bagian perancangan. Perancangan terdiri dari perancangan sistem yang menjelaskan tentang alur sistem yang digunakan secara lebih rinci. Perancangan sistem dilakukan dengan menggunakan *usecase diagram, sequence diagram, activity diagram,* dan *class diagram.* Selain itu pada perancangan sistem terdapat perancangan *database* yang menjelaskan tentang *field* dan *atribut* yang digunakan pada sistem yang dibuat. Selanjutnya perancangan yang dilakukan adalah perancangan tampilan yang menjelaskan tentang tampilan sistem yang akan dibuat.

4.2.1 Perancangan Sistem

Pada perancangan sistem akan memakai model UML yang terdiri dari *usecase diagram, sequence diagram, activity diagram,* dan *class diagram.* Berikut merupakan penjelasan dari perancangan sistem :

4.2.1.1 Usecase Diagram

Usecase Diagram bertujuan untuk mendeskripsikan interaksi antara sistem dan *actor*. Pada *usecase diagram* sistem ini menjelaskan tentang interaksi antara *user* dan sistem yaitu pada proses *login*. Berikut merupakan gambar untuk *usecase diagram* proses *login* :



Gambar 4.3 *Usecase Diagram* Proses Login Aplikasi

Pada *usecase diagram* diatas terdapat interaksi antara *user* dan sistem yang terjadi pada sistem *Single Sign On* (SSO) yang akan dibangun. Usecase *diagram* dibangun dengan 4 proses utama. Yaitu otentikasi, ubah *password*, *switch* aplikasi serta kelola otorisasi.

Adapun *usecase* spesifikasi pada proses otentikasi dapat dilihat pada Tabel 4.1 berikut.

Tabel 4.1 *Usecase* spesifikasi otentikasi

***Use case* : Otentikasi**	
Aktor utama	Operator, Mahasiswa, Dosen, Pegawai, Admin Jurusan
Kondisi awal	Pengguna belum masuk ke dalam sistem
Kondisi akhir	Pengguna berhasil masuk ke dalam sistem
Main success scenario	1. *Use case* dimulai ketika pengguna akan mengakses sistem 2. Pengguna memasukkan data Otentikasi berupa *username* dan *password*. 3. Sistem mengirim data otentikasi kepada SSO 4. SSO melakukan verifikasi terhadap data otentikasi

***Use case* : Otentikasi**	
	5. Data otentikasi terverfikasi 6. Sistem mengarahkan pengguna ke halaman beranda 7. Sistem melakukan segmentasi pada citra latih berdasarkan
Alternative scenario	-

Adapun *usecase* spesifikasi pada proses ubah *password* dapat dilihat pada Tabel 4.2 berikut.

Tabel 4.2 *Usecase* spesifikasi ubah *password*

Use case* : Ubah *Password	
Aktor utama	Operator, Mahasiswa, Dosen, Pegawai, Admin Jurusan
Kondisi awal	Pengguna sudah berada di dalam sistem
Kondisi akhir	Pengguna berhasil mengubah *password*
Main success scenario	1. *Use case* dimulai ketika pengguna akan mengubah *password* 2. Pengguna memasukkan data *password* yang lama 3. Pengguna memasukkan data *password* yang baru 4. Sistem melakukan pengecekan *password* yang lama ke *database server* 5. Sistem mengirim data *password* yang baru ke dalam *database server* *6.* *Password* yang baru berhasil disimpan
Alternative scenario	-

Adapun *usecase* spesifikasi pada proses *switch* aplikasi dapat dilihat pada Tabel 4.3 berikut.

Tabel 4.3 *Usecase* spesifikasi *switch aplikasi*

***Use case* : *Switch* Aplikasi**	
Aktor utama	Operator, Mahasiswa, Dosen, Pegawai, Admin Jurusan
Kondisi awal	Pengguna sudah berada di dalam sistem
Kondisi akhir	Pengguna berhasil pindah aplikasi
Main success scenario	1. *Use case* dimulai ketika pengguna ingin pindah dari satu aplikasi ke aplikasi yang lain 2. Pengguna memilih aplikasi serta modul aplikasi yang akan dituju 3. SSO melakukan pemeriksaan hak akses pengguna terhadap aplikasi yang dituju ke *database server* *4.* Akses tersedia. Pengguna berhasil pindah aplikasi

***Use case* : *Switch* Aplikasi**	
Alternative scenario	-

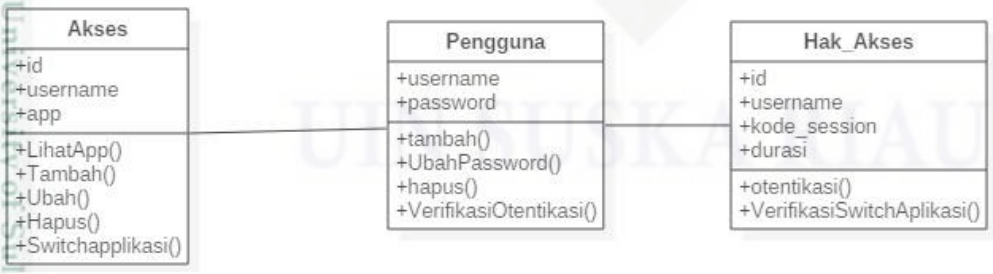
Adapun *usecase* spesifikasi pada proses kelola otorisasi dapat dilihat pada Tabel 4.4 berikut.

Tabel 4.4 *Usecase* spesifikasi kelola otorisasi

***Use case* : Kelola Otorisasi**	
Aktor utama	Operator
Kondisi awal	Operator sudah berada di dalam sistem
Kondisi akhir	Operator berhasil mengelola otorisasi
Main success scenario	1. *Use case* dimulai ketika operator akan mengelola otorisasi 2. Sistem menampilkan data seluruh pengguna 3. Pengguna memilih lihat hak akses 4. Sistem menampilkan hak akses untuk pengguna tertentu 5. Operator mengklik tombol tambah akses. 6. Operator memasukkan data hak akses 7. Hak akses berhasil disimpan
Alternative scenario	-

4.2.1.2 Class Diagram

Class diagram merupakan *diagram* yang menunjukkan kelas-kelas dan hubungan antara masing–masing kelas yang terjadi pada suatu sistem. Pada *class diagram* akan terlihat hubungan antara tabel pada *database* yang digunakan dari sistem tersebut. Berikut merupakan gambar untuk *class diagram* pada sistem ini :



Gambar 4.4 *Class Diagram Database Server*

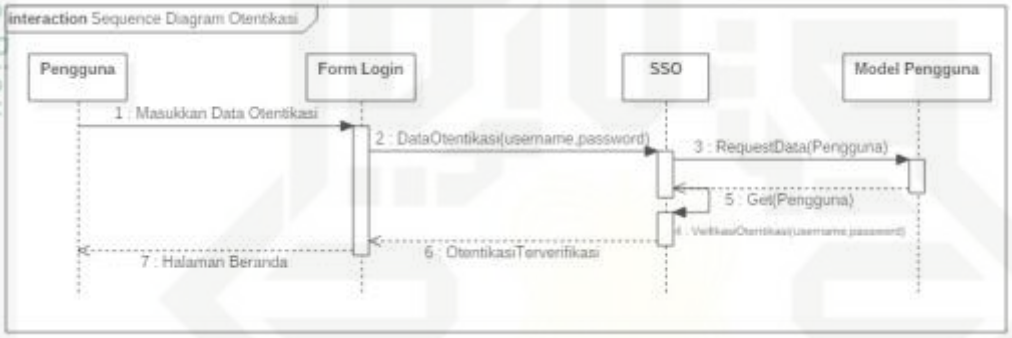
4.2.1.3 Sequence Diagram

Sequence Diagram adalah proses dari tahap ke tahap yang dilakukan untuk menghasilkan suatu proses sesuai dengan *usecae diagram*. Pada sistem ini,

sequence diagram yang didapatkan dari sistem SSO adalah *Sequence Diagram* Otentikasi, *Sequence Diagram* ubah *password*, *Sequence Diagram switch* aplikasi, dan *Sequence Diagram* Kelola Otorisasi ke sistem SSO. Berikut merupakan penjelasan untuk masing – masing *sequence diagram*:

1. *Sequence Diagram* Otentikasi

Sequence diagram otentikasi adalah proses tahapan yang dilakukan *user* saat proses otentikasi dilakukan. Berikut merupakan gambar untuk *sequence diagram* otentikasi :

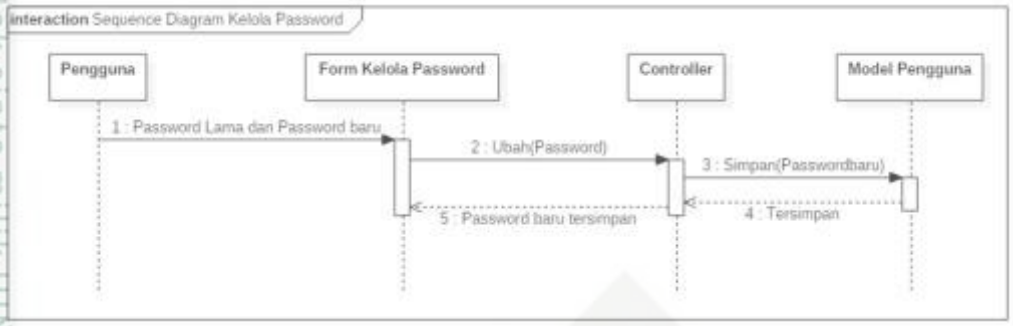


Gambar 4.5 *Sequence Diagram* Otentikasi

Sequence diagram Otentikasi adalah tahapan yang dilakukan saat *user* masuk pada sistem. Pada *sequence diagram* ini, *user* terlebih dahulu mengakses aplikasi. Selanjutnya sistem akan menampilkan halaman *login* untuk *user*. Untuk selanjutnya *user* akan memasukkan *username* dan *password* yang akan dicek oleh SSO pada *database* server. Selanjutnya jika *username* dan *password* ada pada *database* server, sistem akan masuk pada halaman beranda dan *user* dapat mengakses aplikasi sesuai hak aksesnya.

2. *Sequence Diagram* Ubah *password*

Sequence Diagram ubah *password* menjelaskan tentang tahapan yang dilakukan saat proses ubah *password* dilakukan. Proses ubah *password* pada *sequence diagram* ini terjadi pada saat *user* telah berhasil masuk ke sistem setelah proses otentikasi. Berikut merupakan gambar untuk *sequence diagram* ubah *password*:

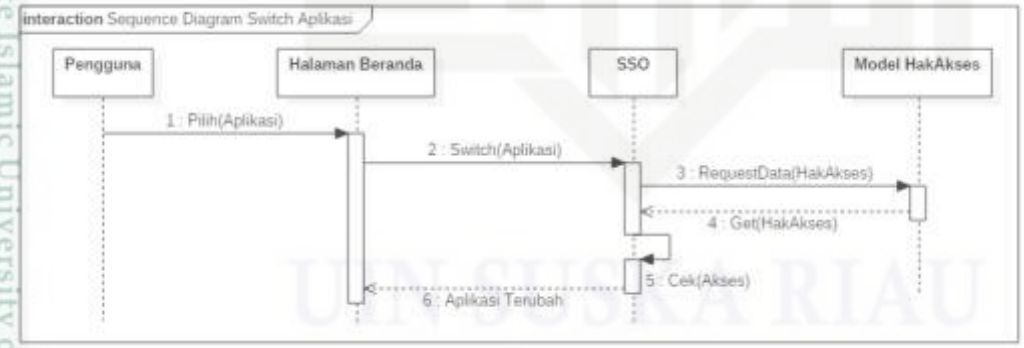


Gambar 4.6 *Sequence Diagram* Ubah *password*

Pada *sequence diagram* diatas, proses terjadi setelah *user* berhasil mengakses aplikasi melalui proses otentikasi. Langkah pertama adalah dengan memilih menu ubah *password*. Kemudian sistem akan menampilkan form ubah *password*. Kemudian, user mengimputkan *password* yang baru untuk kemudian di simpan ke database *server*.

3. Sequence Diagram Switch Aplikasi

Sequence Diagram switch aplikasi menjelaskan tentang tahapan yang dilakukan saat proses switch aplikasi dilakukan. Proses switch aplikasi pada *sequence diagram* ini terjadi pada saat *user* telah berhasil masuk ke sistem setelah proses otentikasi. Berikut merupakan gambar untuk *sequence diagram* switch aplikasi:



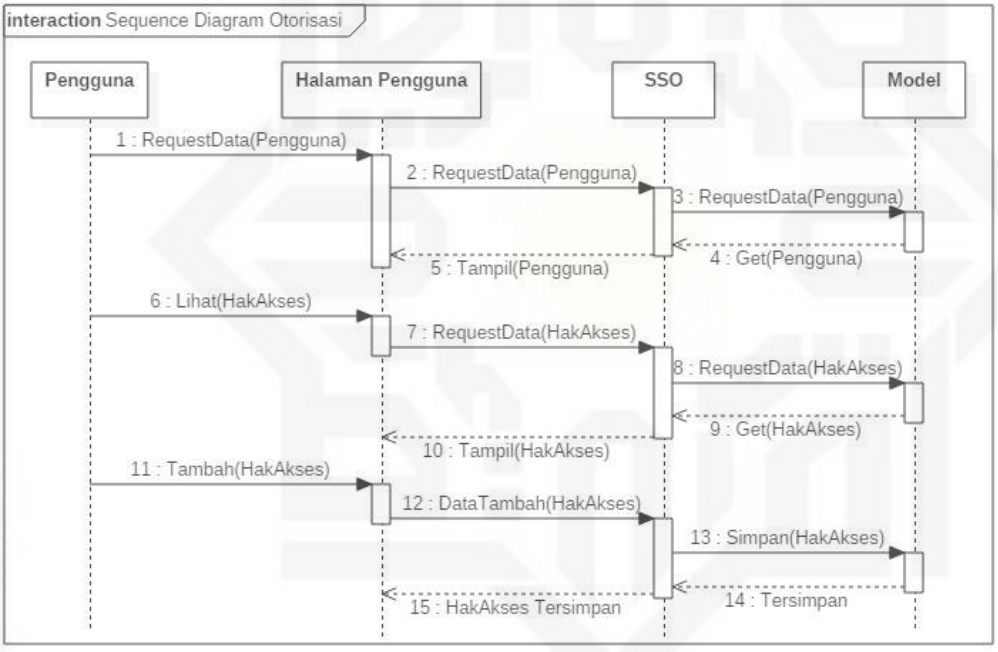
Gambar 4.7 *Sequence Diagram* Switch Aplikasi

Pada *sequence diagram* diatas, proses terjadi setelah *user* berhasil mengakses aplikasi melalui proses otentikasi. Langkah pertama adalah dengan memilih menu apps. Kemudian sistem akan menampilkan daftar akses aplikasi yang tersedia untuk user tersebut. Kemudian, user memilih aplikasi yang akan di

akses. Kemudian SSO melakukan pengecekan terhadap hak akses *user* terhadap aplikasi yang akan di akses. Jika akses tersedia, maka sistem akan pindah ke halaman aplikasi yang dituju.

4. Sequence Diagram Kelola Otorisasi

Sequence Diagram kelola otorisasi menjelaskan tentang tahapan yang dilakukan saat proses kelola otorisasi dilakukan. Proses kelola otorisasi pada *sequence diagram* ini terjadi pada saat *user* telah berhasil masuk ke sistem setelah proses otentikasi. Berikut merupakan gambar untuk *sequence diagram* kelola otorisasi :



Gambar 4.8 *Sequence Diagram* Kelola Otorisasi

Pada *sequence diagram* diatas, proses terjadi setelah *user* berhasil mengakses aplikasi melalui proses otentikasi. Langkah pertama adalah dengan memilih menu pengguna. Kemudian sistem akan menampilkan semua data pengguna yang terdapat di *database server*. Kemudian, user memilih pilihan lihat akses untuk melihat hak akses pada user tertentu. Setelah sistem menampilkan hak akses untuk user tersebut, maka *user* dapat memilih tambah akses untuk dapat menambahkan akses pada *user* yang telah dipilih dengan menginputkan aplikasi

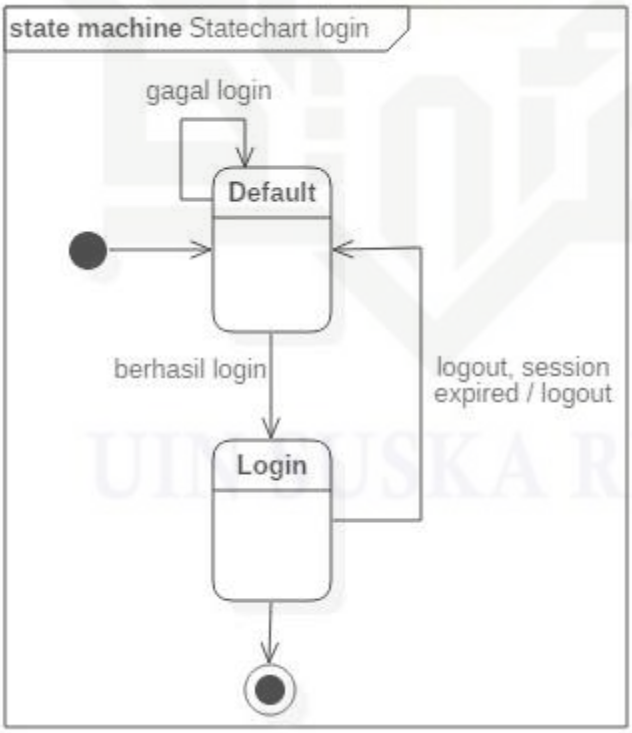
dan modul yang tersedia untuk *user* tersebut. Setelah aplikasi dan modul dipilih, maka akan disimpan ke dalam *database server*.

4.2.1.4 Statechart Diagram

Diagram *statechart* merupakan diagram alir kontrol yang menggambarkan perubahan dari satu kejadian ke kejadian berikutnya dalam suatu sistem. *Statechart diagram* dibawah ini menjelaskan perilaku (*behavior*) dari sistem dalam merespon aksi yang dilakukan oleh *user*.

1. *Statechart Otentikasi*

Pada *statechart* berikut menggambarkan posisi *user* pada awalnya berada pada posisi *default*. Selanjutnya untuk sampai pada proses *login*, *user* harus memasukkan *username* dan *password* yang akan diverifikasi oleh SSO. Dalam pengecekan yang dilakukan SSO, selain *username* dan *password* akan dilakukan pengecekan masa *session* akun tersebut, jika sudah *expired* maka *user* tidak bisa *login.* S*tatechart Otentikasi* yang dilakukan pada penelitian ini dapat dilihat pada Gambar 4.9 berikut:



Gambar 4.9 *Statechart Otentikasi*

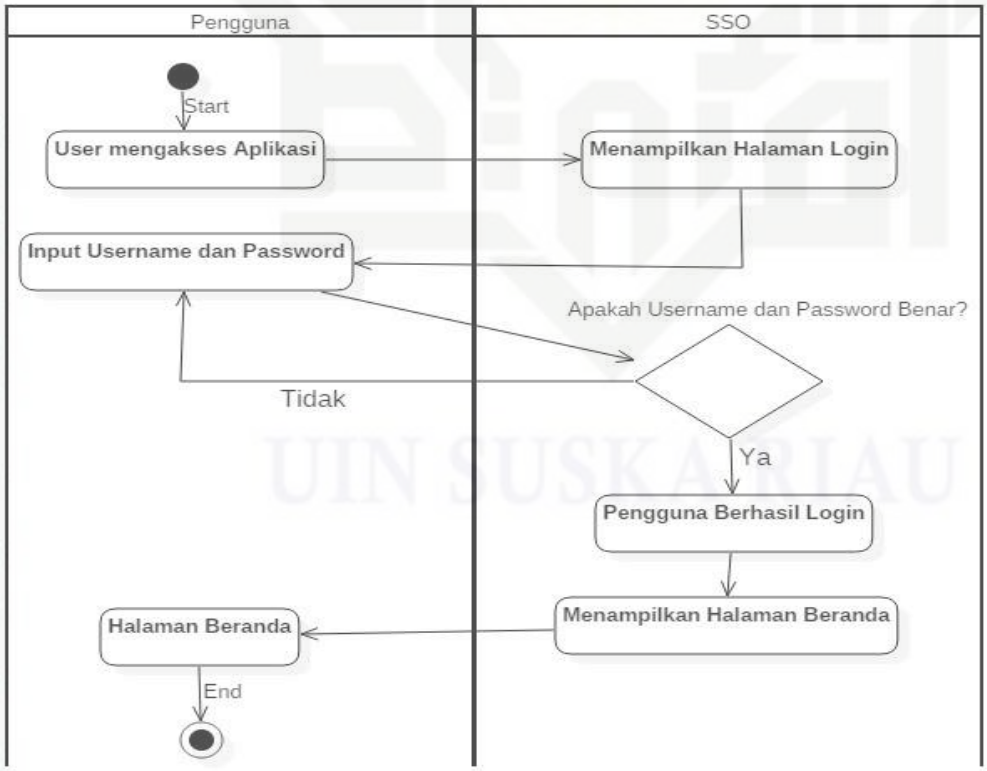
Pada *statechart diagram* diatas, dapat dilihat *state user* sebelum dan sesudah proses otentikasi. Seorang pengguna sistem hasrus melalui proses otentikasi terlebih dahulu sebelum dapat mengakses sistem. Proses otentikasi ini diatur oleh SSO.

4.2.1.5 Activity Diagram

Activity diagram menjelaskan tentang analisa dari sistem dengan membuat model proses dari sistem tersebut. Pada *activity diagram* akan dijelaskan alur dari sistem yang dibuat dimulai dari *user* melakukan proses dari langkah awal sampai selesai. Pada sistem ini terdapat 4 *activity diagram* yang dibutuhkan yaitu *activity diagram* otentikasi, *activity diagram* ubah *password*, *activity diagram switch* aplikasi dan *activity diagram* kelola otorisasi.

1. *Activity Diagram* Otentikasi

Pada *activity diagram Otentikasi*, menjelaskan tentang proses *Otentikasi* yang dilakukan oleh *user* pada saat akan masuk ke dalam sistem. Berikut merupakan gambar untuk *activity diagram Otentikasi*:

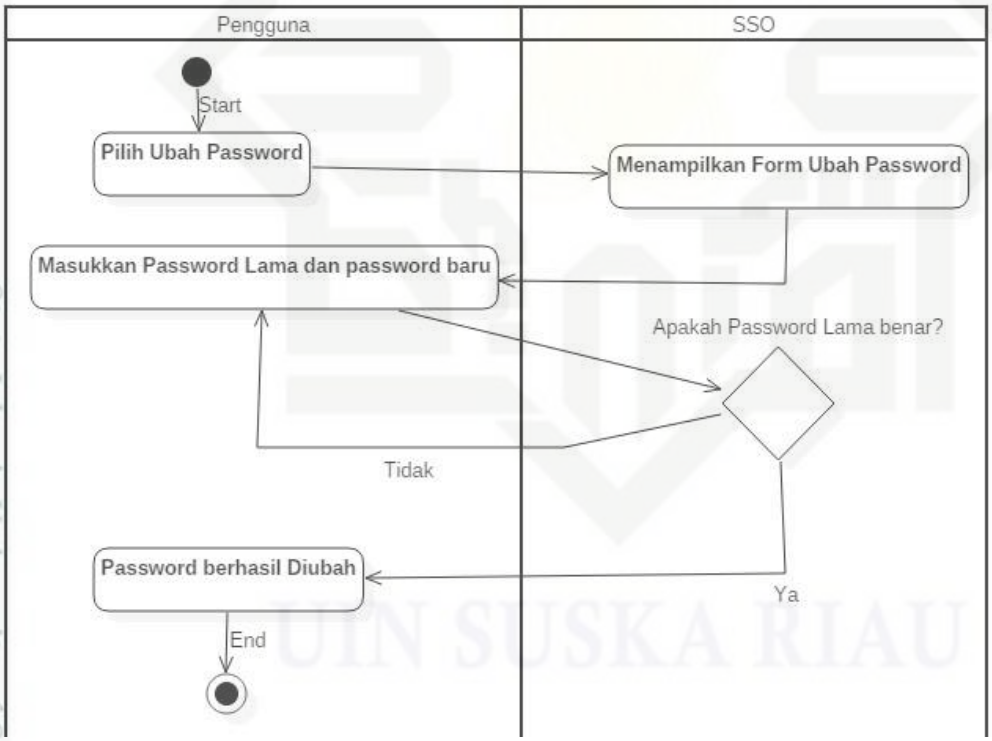


Gambar 4.10 *Activity Diagram Otentikasi*

Pada gambar diatas menjelaskan bahwa untuk *activity diagram otentikasi*, pada saat *user* mengakses aplikasi, untuk masuk kedalam aplikasi *user* diminta untuk memasukkan *username* dan *password*. Sebelumnya aplikasi akan menampilkan halaman otenikasi untuk *user*. Jika *user* berhasil memasukkan *username* dan *password* dengan benar, *user* akan diarahkan ke halaman beranda aplikasi. Namun jika tidak berhasil, *user* akan kembali diminta untuk memasukkan *username* dan *password* dengan benar.

2. *Activity Diagram* Ubah *Password*

Pada *activity diagram* ubah *password,* menjelaskan tentang proses ubah *password* yang dilakukan oleh *user* setelah user berhasil masuk ke dalam sistem. Berikut merupakan gambar untuk *activity diagram* ubah *password:*



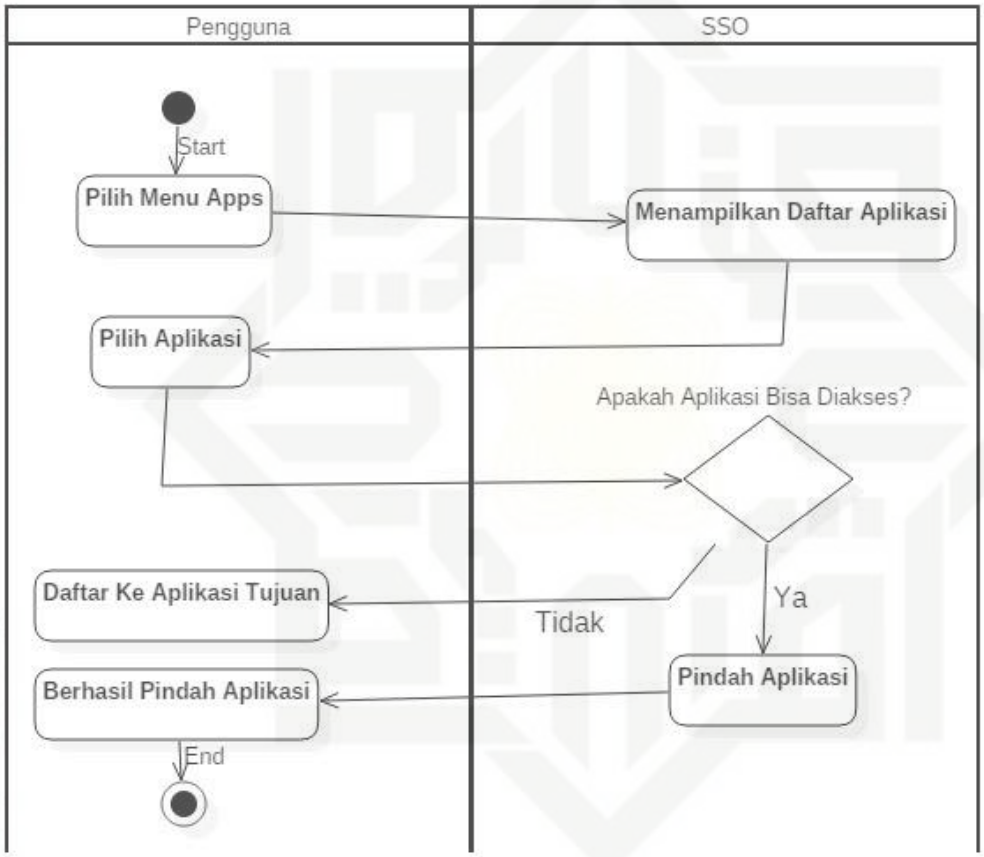
Gambar 4.11 *Activity Diagram* Ubah *Password*

Pada gambar diatas menjelaskan bahwa untuk *activity diagram* ubah *password,* pada saat *user* telah masuk ke aplikasi, *user* dapat memilih ubah

password. Maka selanjutnya sistem akan menampilkan *form* ubah *password*. Kemudian,user menginputkan *password* yang baru dan disimpan ke dalam sistem.

3. *Activity Diagram Switch* Aplikasi

Pada *activity diagram Switch* Aplikasi, menjelaskan tentang proses *switch* aplikasi yang dilakukan oleh *user* setelah user berhasil masuk ke dalam sistem. Berikut merupakan gambar untuk *activity diagram Switch* Aplikasi:

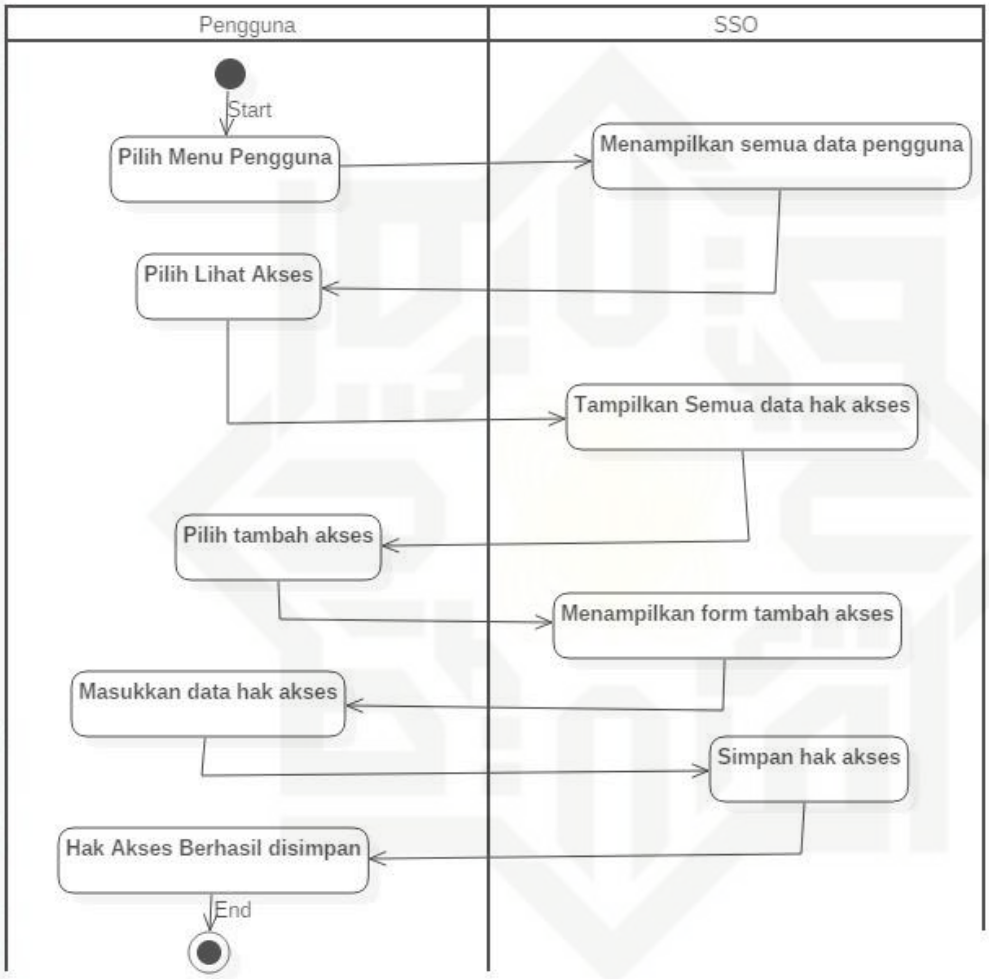


Gambar 4.12 *Activity Diagram Switch* Aplikasi

Pada gambar diatas menjelaskan bahwa untuk *activity diagram switch* aplikasi, pada saat *user* telah masuk ke aplikasi, *user* dapat memilih menu pilih aplikasi untuk memilih aplikasi yang akan di *switch*. Kemudian sistem akan memeriksa akses user tersebut terhadap aplikasi yang akan di *switch*. Jika akses tersedia, maka pengguna akan diarahkan ke halaman beranda aplikasi yang dituju. Jika akses tidak tersedia, maka user tidak dapat mengakses aplikasi yang diinginkan.

4. *Activity Diagram* Otorisasi

Pada *activity diagram O*torisasi, menjelaskan tentang proses *O*torisasi yang dilakukan oleh *user* setelah user berhasil masuk ke dalam sistem. Berikut merupakan gambar untuk *activity diagram Otorisasi*:

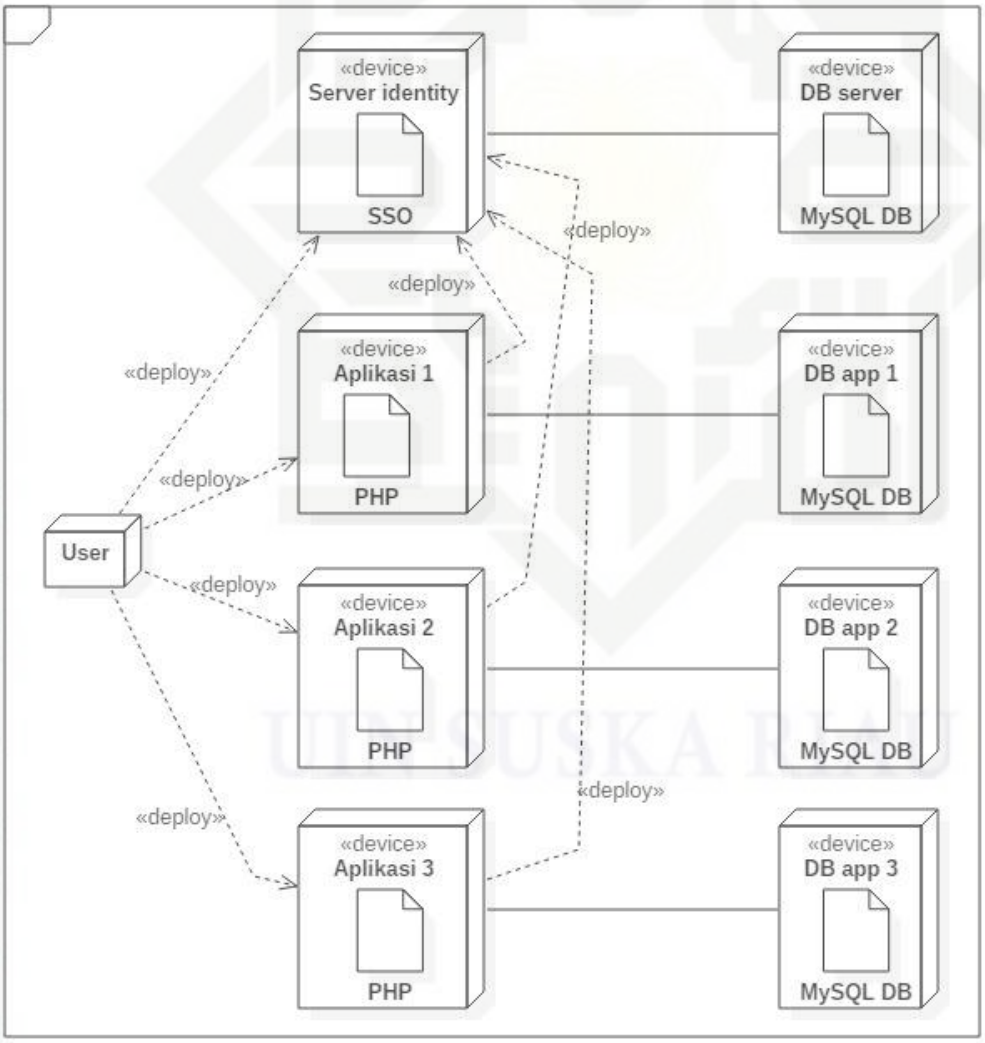


Gambar 4.13 *Activity Diagram Otorisasi*

Pada gambar diatas menjelaskan bahwa untuk *activity diagram otorisasi*, pada saat *user* mengakses aplikasi, *user* dapat memilih menu pengguna. Kemudian sistem akan menampilkan daftar seluruh pengguna untuk di pilih aksesnya. Kemudian user memilih tambah akses, untuk menambahkan akses terhadap user tertentu dengan memilih aplikasi dan modul yang akan disediakan untuk *user* tersebut.

4.2.1.6 Deployment Diagram

Deployment diagram merupakan diagram yang menjelaskan arsitektur fisik dari *hardware* dan *software* sistem yang dikenal dengan istilah *node*. *Deployment diagram* menggambarkan hubungan dari komponen – komponen sistem seperti *software* dan *hardware* yang membentuk arsitektur sistem tersebut. Dalam *deployment diagram* Implementasi Layanan SSO ini terdiri dari 3 aplikasi dan *server* yang digunakan. Dalam mengakses masing-masing sistem tersebut, *user* akan dihubungkan ke server untuk melakukan pengecekan *username, password* dan masa *session* dari akun *user* yang akan *login*. Berikut merupakan gambar *deployment diagram* untuk sistem Implementasi Layanan SSO:



Gambar 4.14 *Deployment Diagram*

Pada *deployment diagram* diatas dapat dilihat bahwa *user* yang telah masuk ke dalam aplikasi dapat mengakses 3 aplikasi yang tersedia selama akses untuk *user* tersebut diberikan. Pemberian akses ini akan diatur dan dikelola oleh SSO.

4.2.2 Perancangan Basis Data

Pada aplikasi SSO yang akan dibangun, terdiri dari 3 perancangan basis data, yaitu perancangan basis data konseptual, perancangan basis data logikal serta perancangan basis data fisikal. Adapun perancangan basis data tersebut dapat dilihat sebagai berikut:

4.2.2.1 Perancangan Basis Data Konseptual

Pada tahap ini, ditentukan siapa saja yang terlibat ke dalam sistem, apa saja input yang diperlukan hingga *output* apa yang diinginkan dari basis data.

1. Pihak yang terlibat dalam sistem (*actor*), yaitu: Dosen, Mahasiswa, Operator serta Admin Jurusan.
2. Input yang diperlukan, yaitu: Data pengguna, data hak akses setiap pengguna data otentikasi setiap pengguna.
3. Output yang diharapkan dari basis data, yaitu: Informasi pengguna, informasi hak akses pengguna, informasi aplikasi pengguna.

4.2.2.2 Perancangan Basis Data Logikal

Pada tahap ini, merupakan tahapa perancangan ERD dengan terlebi dahulu menentukan entitas dan atribut yang terlibat pada basis data, yaitu:

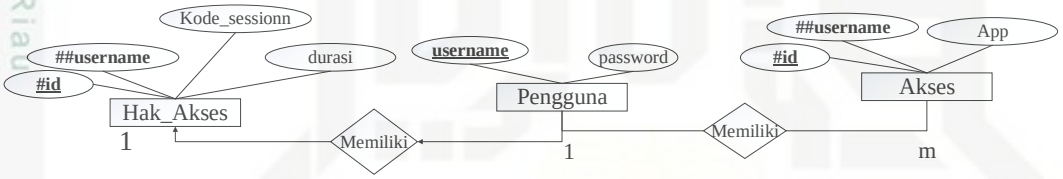
1. Entitas Pengguna, terdiri dari 2 atribut, yaitu: *username* dan *password.*
2. Entitas Akses, terdiri dari 3 atribut, yaitu: id, *username* dan app
3. Entitas HakAkses, terdiri dari 4 atribut, yaitu: id, username, kode_session dan durasi.

Setelah setiap entitas pada basis data ditentukan, maka langkah selanjutnya adalah menetapkan *primary key* untuk setiap entitas seperti yang terlihat berikut:

1. Pengguna ={*username, Password*}
2. Akses ={id, *username, app*}
3. HakAkses = {id, *username*, kode_session, durasi}

4.2.2.3 Perancangan Basis Data Fisikal

Perancangan basis data fisikal merupakan representasi dari perancangan basis data logikal terhadap DBMS yang digunakan sehingga data dapat secara fisik disimpan pada media penyimpanan. Pada penelitian ini, setiap aplikasi memiliki basis data tersendiri untuk mengelola aplikasi. Akan tetapi, dalam proses otentikasi hingga kelola otorisasi seperti yang telah dijelaskan diatas, setiap aplikasi menggunakan basis data *server*. Basis data *server* ini digunakan oleh setiap aplikasi secara bersama-sama. Adapun ERD basis data *server* yang digunakan pada penelitian ini dapat dilihat pada Gambar 4.15 berikut:



Gambar 4.15 ERD Basis Data *Server*

Pada ERD *database server* diatas, terdapat 3 tabel. Yaitu tabel pengguna, akses dan tabel hak_akses. Pada tabel pengguna dengan akses, terdapat relasi *one to many*, Yang artinya 1 pengguna memiliki banyak akses ke berbagai aplikasi yang tersedia. Sedangkan pada tabel pengguna dengan hak_akses, terdapat relasi *one to one*, yang artinya setiap pengguna memiliki 1 hak akses yang berupa kode *session* dan durasi *session*. Hak akses ini dapat digunkan untuk mengakses semua apikasi yang diizinkan untuk pengguna tersebut selama *session* pengguna tersebut masih aktif.

Adapun keterangan dari tabel-tabel pada basis data server dapat dilihat sebagai berikut:

Tabel 4.5 Tabel Pengguna

Nama field	Type dan length	Keterangan
username	*Varchar (50)*	*Username* pengguna
password	*Varchar (200)*	*Password* Pengguna

Pada Tabel 4.5 diatas, diketahui bahwa tabel pengguna berisi data pengguna yang terdapat pada aplikasi yang akan dibangun. Tabel pengguna digunakan untuk proses otentikasi serta ubah *password*. Tabel pengguna memiliki 2 entitas, yaitu: *username* serta *password*.

Tabel 4.6 Tabel Akses

Nama field	Type dan length	Keterangan
id	*Integer(11)*	Kode hak akses
username	*Varchar (50)*	*Username* pengguna (*foreign key*)
app	*Varchar (50)*	Aplikasi yang dapat diakses Pengguna

Pada Tabel 4.6 diatas, diketahui bahwa tabel akses berisi data aplikasi yang dapat diakses untuk setiap pengguna yang terdapat pada aplikasi yang akan dibangun. Tabel akses digunakan untuk proses *switch* aplikasi serta kelola otorisasi. Tabel hak_akses memiliki 3 entitas, yaitu: id, *username* serta app.

Tabel 4.7 Tabel Hak_akses

Nama field	Type dan length	Keterangan
Id	*Integer(11)*	Kode hak akses
username	*Varchar (50)*	*Username* pengguna (*foreign key*)
Kode_session	*Varchar (500)*	Kode *Credential* pengguna
Durasi	*Integer (11)*	Durasi *Credential* berlangsung

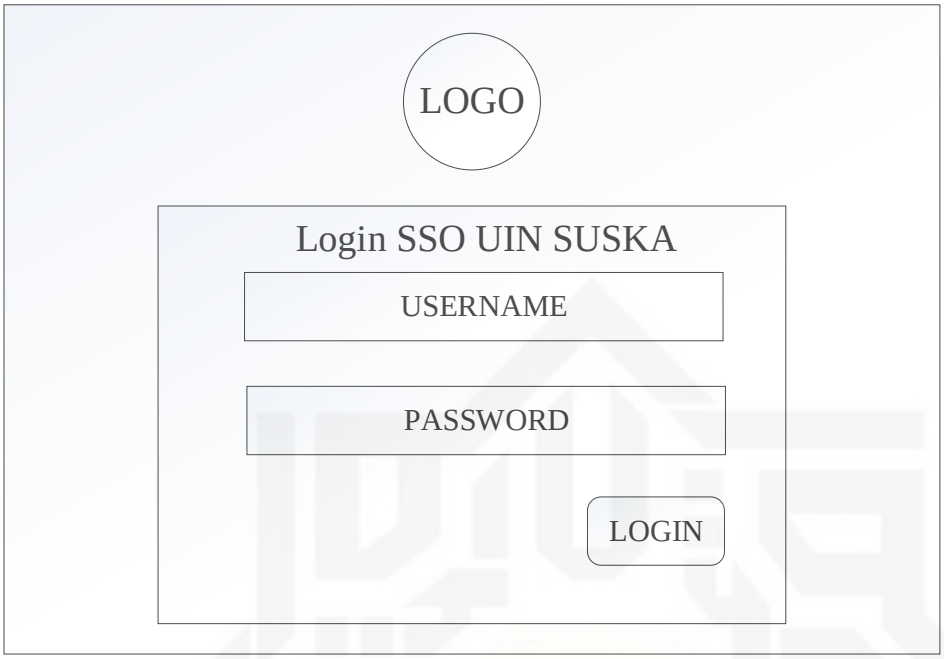
Pada Tabel 4.7 diatas, diketahui bahwa tabel hak_akses berisi data hak akses untuk setiap pengguna yang terdapat pada aplikasi yang akan dibangun. Tabel hak_akses digunakan untuk proses otentikasi serta *switch* aplikasi. Tabel hak_akses memiliki 4 entitas, yaitu: id, *username*, kode_session serta durasi.

4.2.3 Perancangan Antarmuka (*Interface*)

Perancangan antarmuka (*interface*) merupakan perancangan tampilan aplikasi yang akan dibangun yang selanjutnya akan digunakan sebagai acuan dalam melakukan implementasi. Secara umum, perancangan antarmuka pada aplikasi SSO pada aplikasi akademik terpadu UIN SUSKA Riau terdiri dari antarmuka halaman otentikasi, antarmuka halaman ubah *password*, antarmuka halaman *switch* aplikasi dan antarmuka halaman kelola otorisasi. Berikut adalah rancangan antarmuka sistem yang dibangun:

4.2.3.1 Antarmuka Halaman Otentikasi

Pada sistem yang akan dibangun, proses otentikasi dilakukan dengan menggunakan SSO, yaitu semua aplikasi mengakses halaman otentikasi yang sama. Adapun perancangan antarmuka untuk halaman otentikasi pada aplikasi berbasis SSO yang dibangun dapat dilihat pada Gambar 4.16 berikut:



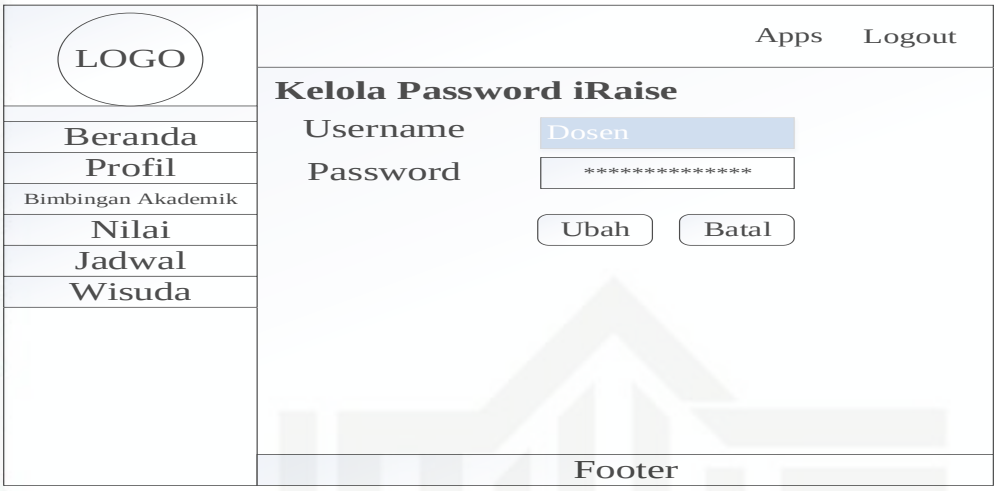
Gambar 4.16 Antarmuka otentikasi

Pada Gambar 4.16 diatas, dapat dilihat bahwa pada proses otentikasi untuk setiap aplikasi. Ketika pengguna akan mengakses aplikasi tertentu, sistem akan mengarahkan pengguna ke halaman otentikasi. Selanjutnya pengguna akan memasukkan *username* dan *password* untuk dapat mengakses aplikasi yang dituju. Jika pengguna tersebut memiliki akses terhadap aplikasi tujuannya, maka pengguna akan diarahkan ke halaman beranda aplikasi tersebut.

Setelah pengguna berhasil *login* ke dalam aplikasi, maka pengguna dapat melakukan proses ubah *password*, *switch* aplikasi serta kelola otorisasi untuk semua aplikasi. Adapun perancangan antarmuka proses-proses tersebut, dapat dilihat sebagai berikut:

4.2.3.2 Antarmuka Halaman Ubah *password* iRaise

Perancangan antarmuka untuk halaman ubah *password* pada aplikasi iRaise yang dibangun dapat dilihat pada Gambar 4.17 berikut:

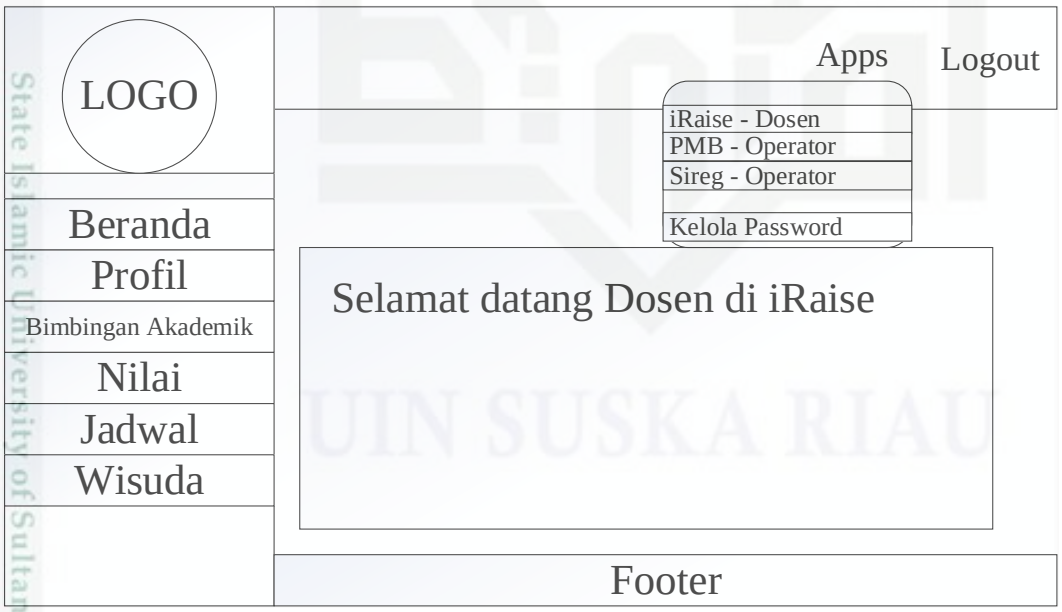


Gambar 4.17 Antarmuka Ubah *password* iRaise

Pada Gambar 4.17 diatas, dapat dilihat bahwa pada ubah *password* aplikasi iRaise, pengguna dapat mengubah *password* yang akan digunakan pada proses otentikasi untuk setiap aplikasi yang diberi akses untuk pengguna tersebut.

4.2.3.3 Antarmuka Halaman *Switch* Aplikasi iRaise

Perancangan antarmuka untuk halaman *switch* aplikasi pada aplikasi iRaise yang dibangun dapat dilihat pada Gambar 4.18 berikut:

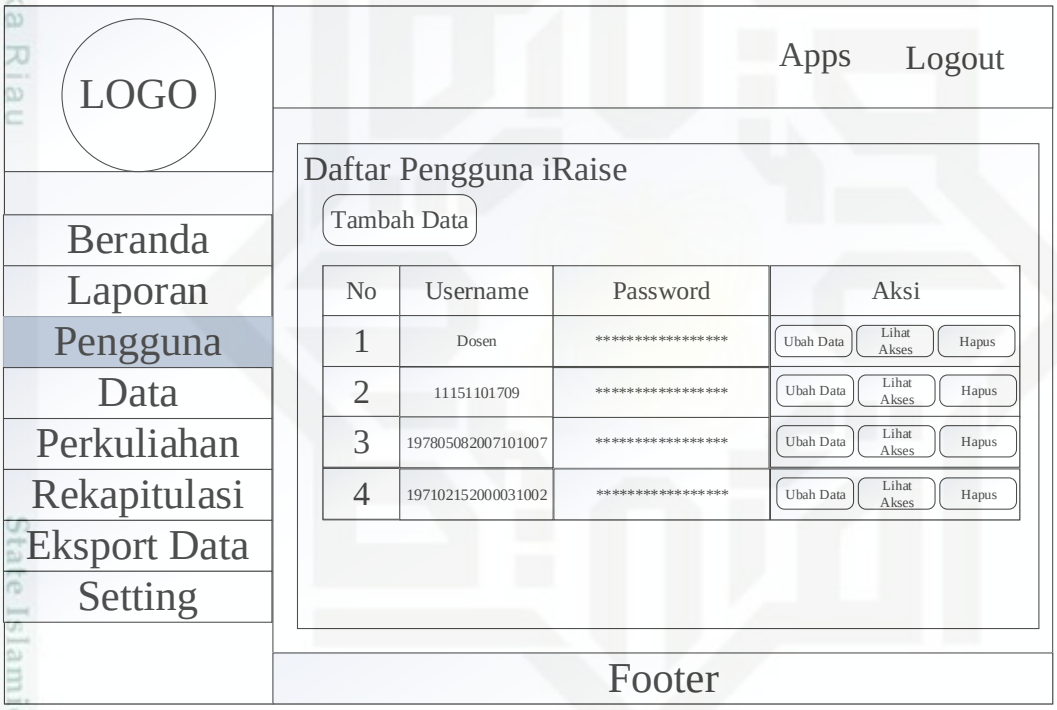


Gambar 4.18 Antarmuka *Switch* Aplikasi iRaise

Pada Gambar 4.18 diatas, dapat dilihat bahwa pada proses *switch* aplikasi iRaise, dilakukan dari menu apps. Kemudian pengguna akan memilih aplikasi yang akan dituju berikutnya sesuai dengan akses yang telah diberikan oleh sistem.

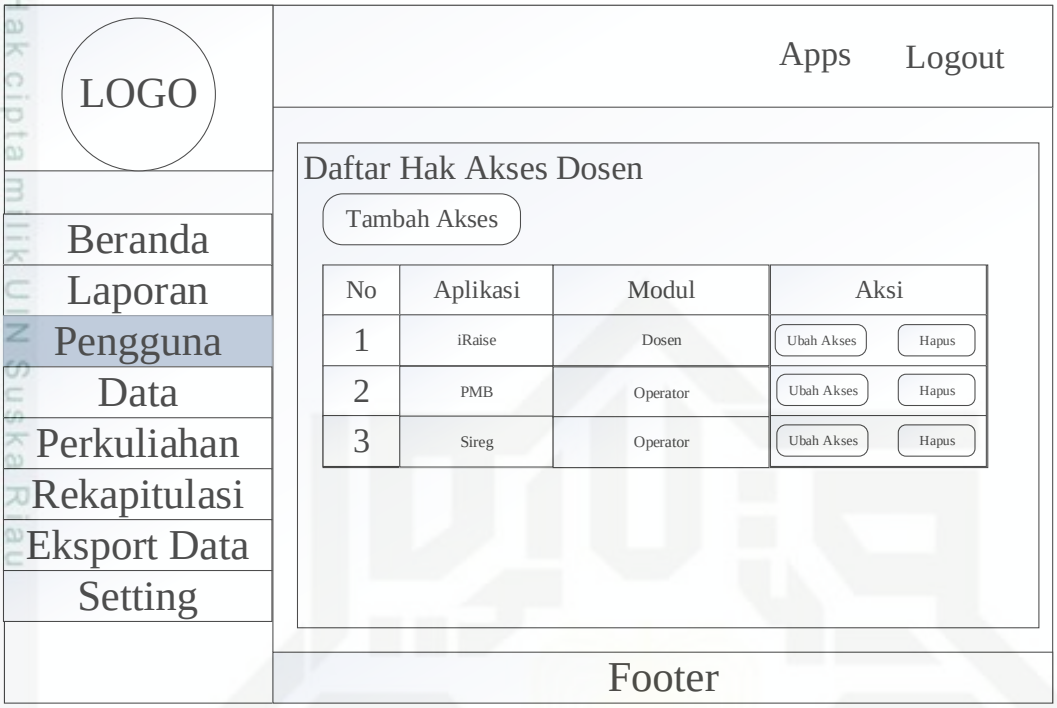
4.2.3.4 Antarmuka Halaman Kelola Otorisasi iRaise

Perancangan antarmuka untuk halaman kelola otorisasi aplikasi pada aplikasi iRaise hanya dapat dilakukan oleh pengguna dengan level operator. Untuk melakukan kelola otorisasi, operator terlebih dahulu masuk ke menu kelola pengguna seperti yang dapat dilihat pada Gambar 4.19 berikut:



Gambar 4.19 Antarmuka Kelola Pengguna iRaise

Pada Gambar 4.19 diatas, dapat dilihat tampilan halaman kelola pengguna. Kelola pengguna terdiri dari tambah data, ubah data, lihat akses serta hapus data pengguna. Untuk mengelola otorisasi terhadap pengguna tertentu, maka operator akan memilih menu lihat akses. Setelah menu lihat akses dipilih, maka akan muncul tampilan kelola otorisasi seperti yang terlihat pada Gambar 4.20 berikut:

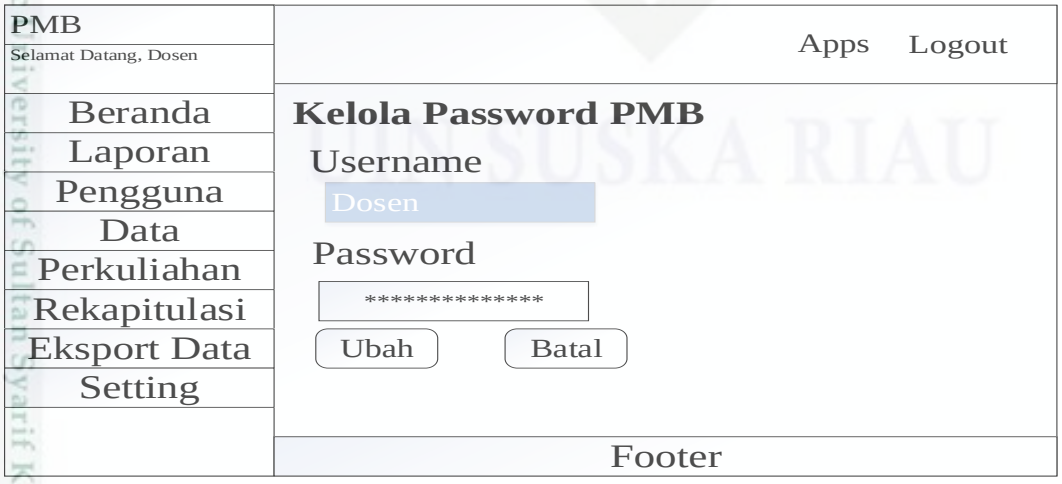


Gambar 4.20 Antarmuka Kelola Otorisasi Pengguna iRaise

Berdasarkan Gambar 4.20 diatas, dapat dilihat bahwa operator dapat mengelola otorisasi untuk pengguna tersebut, yaitu dengan melakukan tambah akses, ubah akses serta hapus akses. Pengaturan akses ini berisi aplikasi dan modul apa saja yang diizinkan untuk pengguna tersebut.

4.2.3.5 Antarmuka Halaman Ubah *password* PMB

Perancangan antarmuka untuk halaman ubah *password* pada aplikasi PMB yang dibangun dapat dilihat pada Gambar 4.21 berikut:

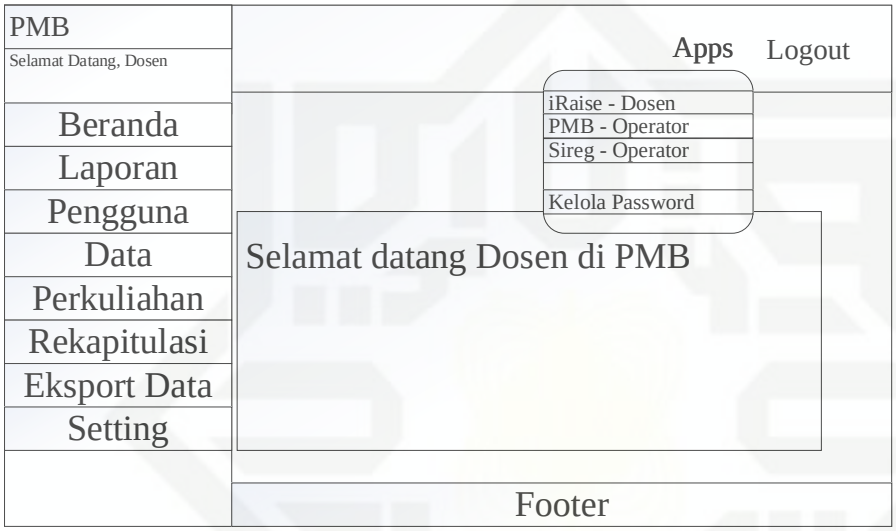


Gambar 4.21 Antarmuka Ubah *password* PMB

Pada Gambar 4.21 diatas, dapat dilihat bahwa pada ubah *password* aplikasi PMB, pengguna dapat mengubah *password* yang akan digunakan pada proses otentikasi untuk setiap aplikasi yang diberi akses untuk pengguna tersebut.

4.2.3.6 Antarmuka Halaman *Switch* Aplikasi PMB

Perancangan antarmuka untuk halaman *switch* aplikasi pada aplikasi PMB yang dibangun dapat dilihat pada Gambar 4.22 berikut:

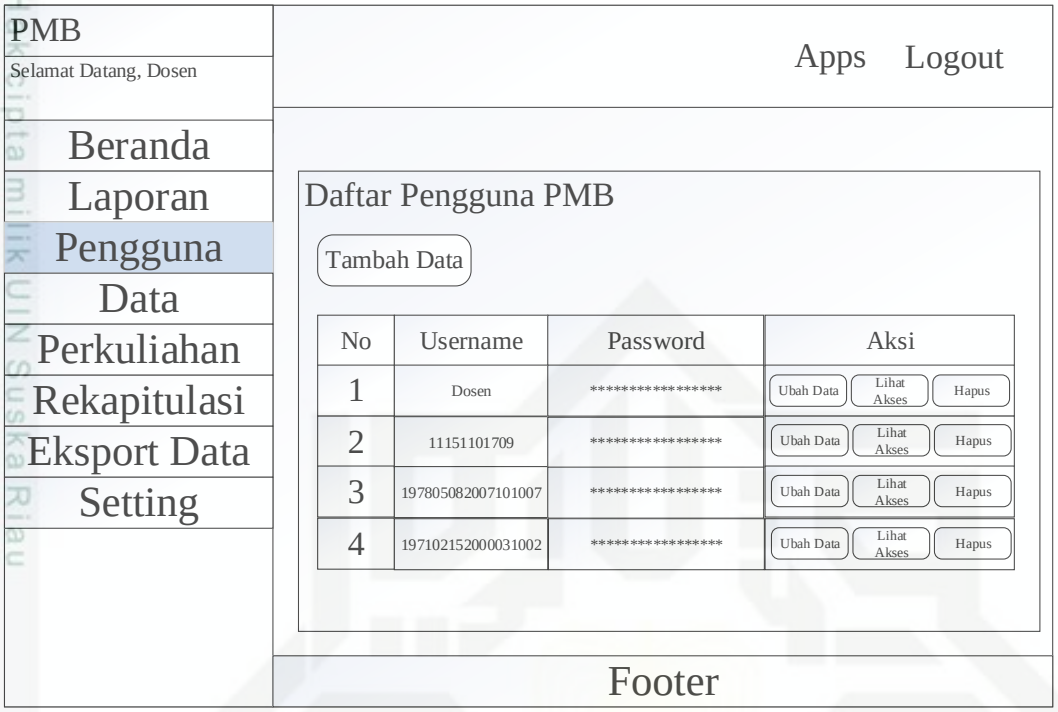


Gambar 4.22 Antarmuka *Switch* Aplikasi PMB

Pada Gambar 4.22 diatas, dapat dilihat bahwa pada proses *switch* aplikasi PMB, dilakukan dari menu *apps*. Kemudian pengguna akan memilih aplikasi yang akan dituju berikutnya sesuai dengan akses yang telah diberikan oleh sistem.

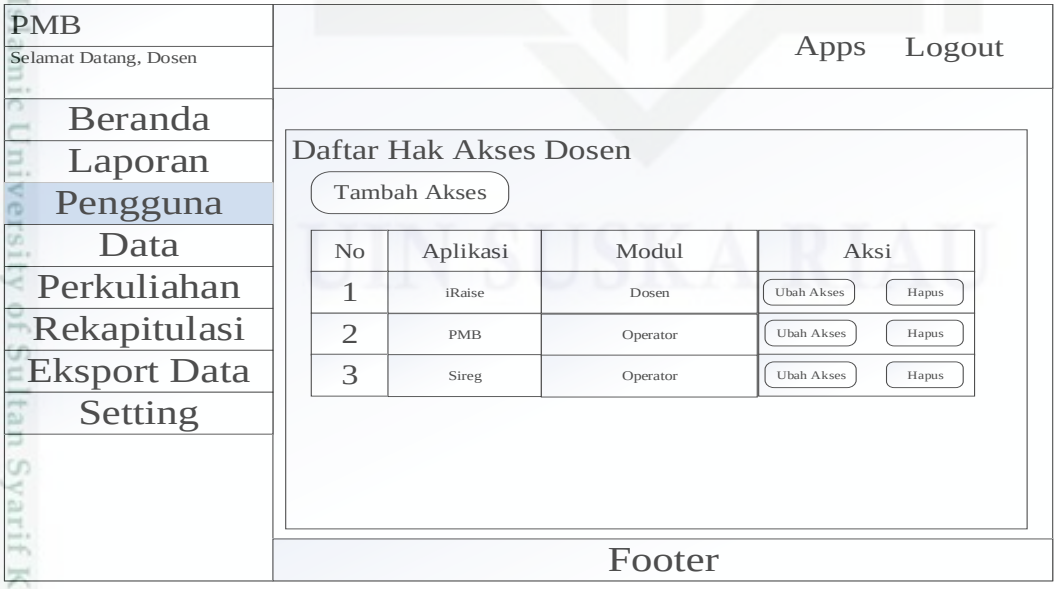
4.2.3.7 Antarmuka Halaman Kelola Otorisasi PMB

Perancangan antarmuka untuk halaman kelola otorisasi aplikasi pada aplikasi PMB hanya dapat dilakukan oleh pengguna dengan level operator. Untuk melakukan kelola otorisasi, operator terlebih dahulu masuk ke menu kelola pengguna seperti yang dapat dilihat pada Gambar 4.23 berikut:



Gambar 4.23 Antarmuka Kelola Pengguna PMB

Pada Gambar 4.23 diatas, dapat dilihat tampilan halaman kelola pengguna. Kelola pengguna terdiri dari tambah data, ubah data, lihat akses serta hapus data pengguna. Untuk mengelola otorisasi terhadap pengguna tertentu, maka operator akan memilih menu lihat akses. Setelah menu lihat akses dipilih, maka akan muncul tampilan kelola otorisasi seperti yang terlihat pada Gambar 4.24 berikut:



Gambar 4.24 Antarmuka Kelola Otorisasi Pengguna PMB

Berdasarkan Gambar 4.24 diatas, dapat dilihat bahwa operator dapat mengelola otorisasi untuk pengguna tersebut, yaitu dengan melakukan tambah akses, ubah akses serta hapus akses. Pengaturan akses ini berisi aplikasi dan modul apasaja yang diizinkan untuk pengguna tersebut.

4.2.3.8 Antarmuka Halaman Ubah *password* Sireg

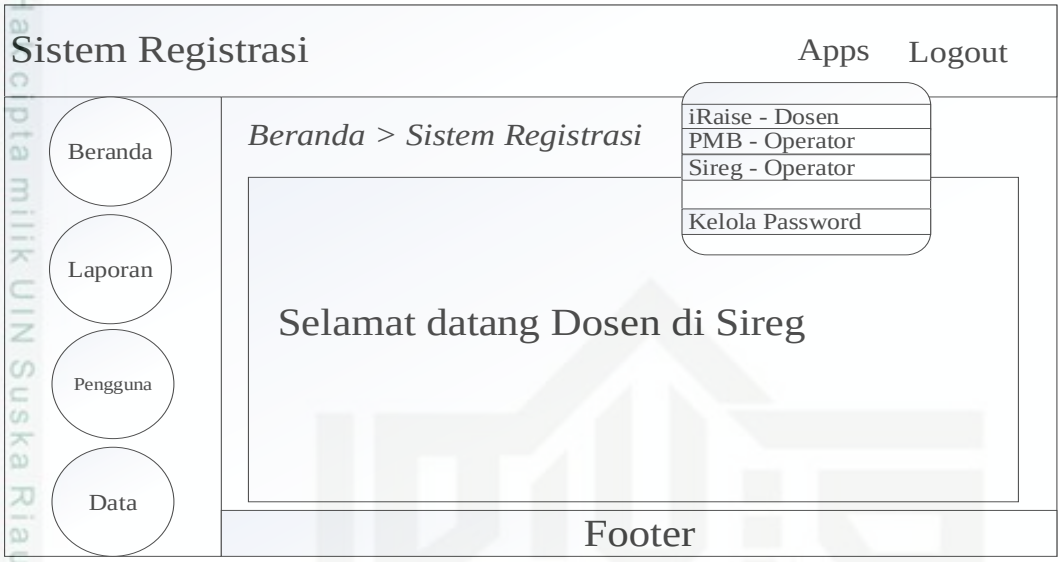
Perancangan antarmuka untuk halaman ubah *password* pada aplikasi Sireg yang dibangun dapat dilihat pada Gambar 4.25 berikut:

Gambar 4.25 Antarmuka Ubah *password* Sireg

Pada Gambar 4.25 di atas, dapat dilihat bahwa pada ubah *password* aplikasi Sireg, pengguna dapat mengubah *password* yang akan digunakan pada proses otentikasi untuk setiap aplikasi yang diberi akses untuk pengguna tersebut.

4.2.3.9 Antarmuka Halaman *Switch* Aplikasi Sireg

Perancangan antarmuka untuk halaman *switch* aplikasi pada aplikasi Sireg yang dibangun dapat dilihat pada Gambar 4.26 berikut:

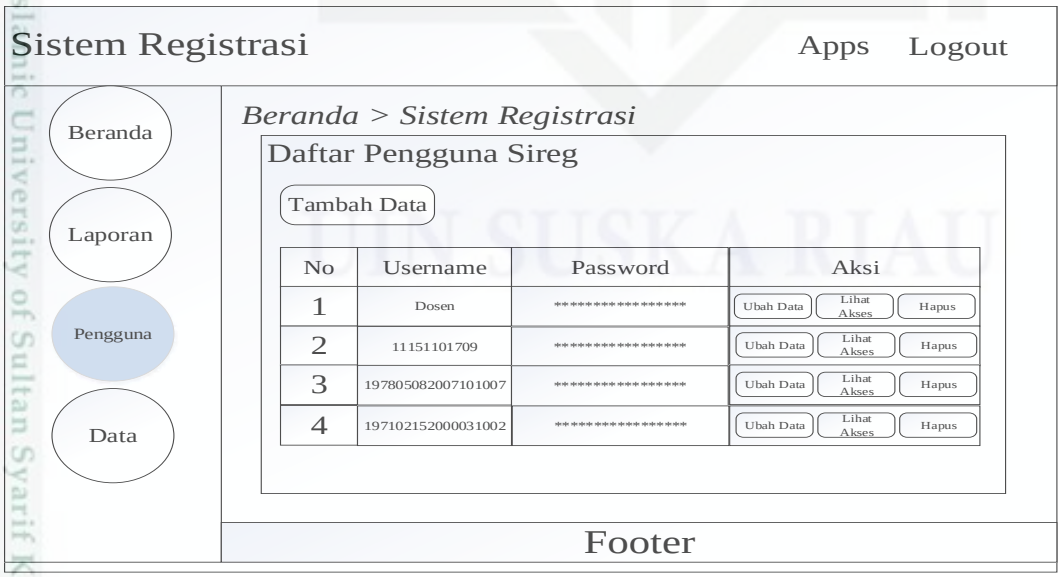


Gambar 4.26 Antarmuka *Switch* Aplikasi Sireg

Pada Gambar 4.26 diatas, dapat dilihat bahwa pada proses *switch* aplikasi Sireg, dilakukan dari menu apps. Kemudian pengguna akan memilih aplikasi yang akan dituju berikutnya sesuai dengan akses yang telah diberikan oleh sistem.

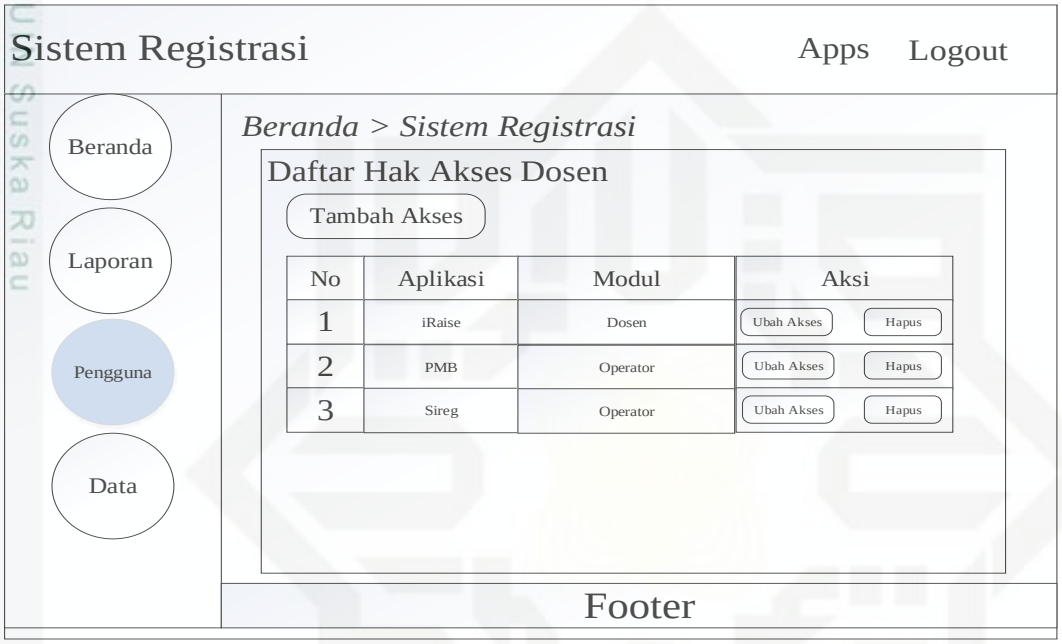
4.2.3.10 Antarmuka Halaman Kelola Otorisasi Sireg

Perancangan antarmuka untuk halaman kelola otorisasi aplikasi pada aplikasi Sireg hanya dapat dilakukan oleh pengguna dengan level operator. Untuk melakukan kelola otorisasi, operator terlebih dahulu masuk ke menu kelola pengguna seperti yang dapat dilihat pada Gambar 4.27 berikut:



Gambar 4.27 Antarmuka Kelola Pengguna Sireg

Pada Gambar 4.27 diatas, dapat dilihat tampilan halaman kelola pengguna. Kelola pengguna terdiri dari tambah data, ubah data, lihat akses serta hapus data pengguna. Untuk mengelola otorisasi terhadap pengguna tertentu, maka operator akan memilih menu lihat akses. Setelah menu lihat akses dipilih, maka akan muncul tampilan kelola otorisasi seperti yang terlihat pada Gambar 4.28 berikut:



Gambar 4.28 Antarmuka Kelola Otorisasi Pengguna Sireg

Berdasarkan Gambar 4.28 diatas, dapat dilihat bahwa operator dapat mengelola otorisasi untuk pengguna tersebut, yaitu dengan melakukan tambah akses, ubah akses serta hapus akses. Pengaturan akses ini berisi aplikasi dan modul apasaja yang diizinkan untuk pengguna tersebut.